A sound formalization of void safety

Alexander Kogtenkov

Schaffhausen Institute of Technology

2021-12-02

Problem

expr.method (args);

Problem

expr.method(args);

Problem

expr.method (args);

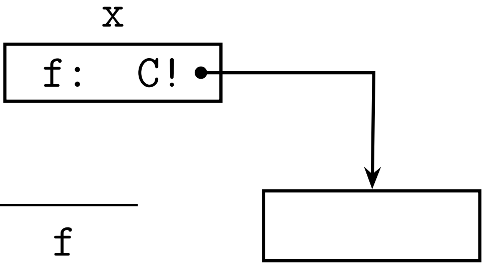
Problem

```
expr.method (args);
```

```
tmp = expr;
if (tmp == null)
    throw new NullPointerException ();
else
    tmp.method(args);
```

Assignment rule

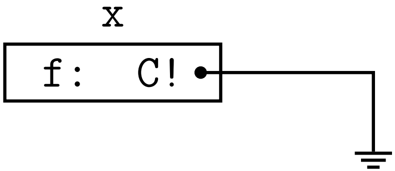
```
y = x.f;
```



			f	
			?	!
y	?	nullable	✓	✓
	!	non-null	✗	✓

Assignment rule

```
y = x.f;
```

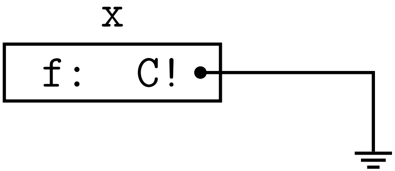


			f	
			?	!
y	?	nullable	✓	✓
	!	non-null	✗	✓

Breaks for new objects!

Object creation

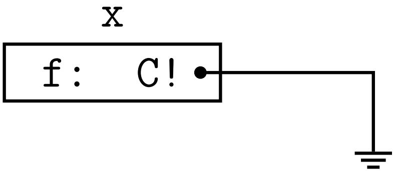
	x.f		f	
			!	?
	1	committed	1!	1?
x	0	free	◇?	◇?
	◇	unclassified	◇?	◇?



Alexander J. Summers, Peter Müller. Freedom Before Commitment. OOPSLA'11

Object creation

		x.f	f	
			!	?
	1	committed	1!	1?
x	0	free	◇?	◇?
	◇	unclassified	◇?	◇?



Alexander J. Summers, Peter Müller. Freedom Before Commitment. OOPSLA’11

Object creation

		x.f	f	
			!	?
x	**1**	committed	**1!**	**1?**
	0	free	◇**?**	◇**?**
	◇	unclassified	◇**?**	◇**?**

	x.f = y;	y		
		1	**0**	◇
x	**1**	✓	✗	✗
	0	✓	✓	✓
	◇	✓	✗	✗

Alexander J. Summers, Peter Müller. Freedom Before Commitment. OOPSLA’11

Proof on paper

	Original	Verified
Premises	$D \sqsubseteq C$	$C \sqsubseteq D$
	—	$m \in methods\ D$
	$mSig\ (D,\ m) = (\gamma,\ S_i,\ S,\ S_j)$	
	$mSig\ (C,\ m) = (\gamma',\ S_i',\ S',\ S_l)$	
	—	$\vdash_{mS} (\gamma,\ S_i,\ S,\ S_j)$
	—	$\vdash_{mS} (\gamma',\ S_i',\ S',\ S_l)$
	$instance\ \vartheta\ S\ (D^{\gamma}! \cdot S_i)$	
Conclusions	$\vartheta' \in instances\ S'\ (C^{\gamma'}! \cdot S_i')$	
	$\vartheta'\ \gamma' = \vartheta\ \gamma'$	
	$\vartheta'\ S_i' = \vartheta\ S_i'$	
	$\vartheta'\ S' = \vartheta\ S$	$\vartheta'\ S' \sqsubseteq \vartheta\ S$
	$\vartheta'\ S_l = \vartheta\ S_l$	

Outline

Type system

Heap properties

Semantics

Proof

Classes

$$D \sqsubseteq C$$

Descendant

Ancestor

Classes

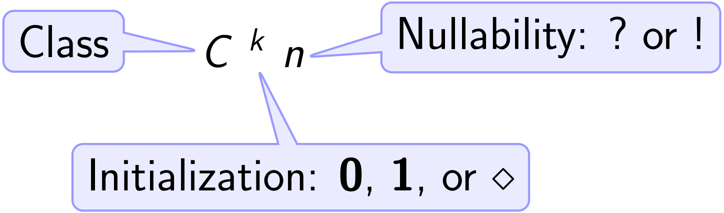
$$D \sqsubseteq C$$

Descendant

Ancestor

$$\mathit{methods}\ C \subseteq \mathit{methods}\ D$$
$$\mathit{fields}\ C \subseteq \mathit{fields}\ D$$

Types



Class

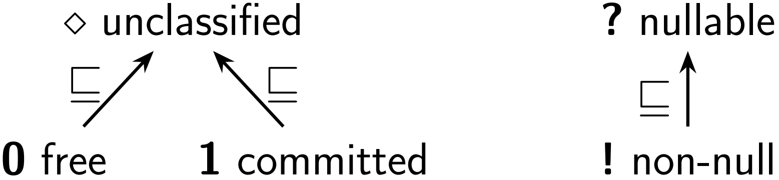
$C^{k}\ n$

Nullability: ? or !

Initialization: $\mathbf{0}$, $\mathbf{1}$, or $\diamond$

Types

$$C^{k}\ n$$



$\diamond$ unclassified

$\sqsubseteq$ $\sqsubseteq$

$\mathbf{0}$ free

$\mathbf{1}$ committed

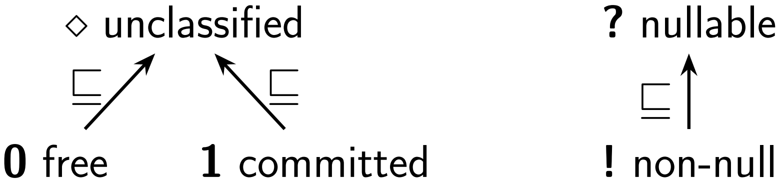
$\mathbf{?}$ nullable

$\sqsubseteq$

$\mathbf{!}$ non-null

Types

$$C^{\,k}\; n$$



$\diamond$ unclassified

$\sqsubseteq$ $\sqsubseteq$

0 free **1** committed

? nullable

$\sqsubseteq$

! non-null

$$C_1^{k_1} n_1 \sqsubseteq C_2^{k_2} n_2 \equiv C_1 \sqsubseteq C_2 \wedge k_1 \sqsubseteq k_2 \wedge n_1 \sqsubseteq n_2$$

Types

Type: $\boxed{C^{\,k}\; n}$

Field type: $\boxed{C\; n}$

$$\begin{tikzcd}
 & \diamond\ \text{unclassified} & \\
\mathbf{0}\ \text{free} \arrow[ur, "\sqsubseteq"] & & \mathbf{1}\ \text{committed} \arrow[ul, "\sqsubseteq"']
\end{tikzcd}$$

$$\begin{tikzcd}
\mathbf{?}\ \text{nullable} \\
\mathbf{!}\ \text{non-null} \arrow[u, "\sqsubseteq"]
\end{tikzcd}$$

$$C_1^{k_1} n_1 \sqsubseteq C_2^{k_2} n_2 \equiv C_1 \sqsubseteq C_2 \wedge k_1 \sqsubseteq k_2 \wedge n_1 \sqsubseteq n_2$$

Signatures

$$fType\ (C,\ f) = T$$

Field type: $C\ \ n$

Signatures

Arguments

Local variables

$$fType\ (C,\ f) = T$$

$$mSig\ (C,\ m) = (\gamma,\quad X_i,\quad S,\quad Y_j)$$

Target initialization

Result

Signatures

$$fType\ (C,\ f) = T$$

$$mSig\ (C,\ m) = (\gamma,\quad X_i,\quad S,\quad Y_j)$$

$$cSig\ C = (X_i,\quad Y_j)$$

Signatures

$$fType\ (C,\ f) = T$$

$$mSig\ (C,\ m) = (\gamma,\quad X_i,\quad S,\quad Y_j)$$

$$cSig\ C = (X_i,\quad Y_j)$$

nonvariant	T
contravariant	$\gamma,\ X_i$
covariant	S

Expressions and statements

$$e ::= x \mid x.f \mid \mathsf{null}$$

Expressions and statements

$e ::= x \mid x.f \mid \mathsf{null} \qquad \text{including } x = \mathsf{res} \text{ and } x = \mathsf{this}$

Expressions and statements

$e ::= x \mid x.f \mid \mathsf{null}$ including $x = \mathsf{res}$ and $x = \mathsf{this}$

$$
\begin{array}{rl}
s ::= & x := e \\
\mid & z.f := y \\
\mid & x := y.m\ (z_i) \\
\mid & x := \mathsf{new}\ C\ (z_i) \\
\mid & x := (t)\ y \\
\mid & s_1;\ s_2
\end{array}
$$

Expressions and statements

$$e ::= x \mid x.f \mid \textsf{null} \qquad \text{including } x = \textsf{res} \text{ and } x = \textsf{this}$$

$$\begin{array}{rl} s ::= & x := e \\ \mid & z.f := y \\ \mid & x := y.m\,(z_i) \\ \mid & x := \textsf{new}\ C\,(z_i) \\ \mid & x := (t)\ y \\ \mid & s_1;\ s_2 \end{array}$$

$\vdash_s s$

Well-formed $\approx x \neq \textsf{this}$

Well-formed signatures

$$\begin{aligned}
\vdash_{mS} (\gamma,\ X_i,\ S,\ Y_j) \;\; \equiv \;\; & \mathit{vars}\ (S \cdot\ Y_j) \subseteq \mathit{var}\ \gamma \cup \mathit{vars}\ X_i \\
\wedge \;\; & \mathrm{this} \notin X_i \cup\ Y_j \\
\wedge \;\; & \mathrm{res} \notin X_i \cup\ Y_j \\
\wedge \;\; & \mathit{distinct}\ (X_i\ @\ Y_j)
\end{aligned}$$

Well-formed signatures

$$\begin{aligned}
\vdash_{mS} (\gamma, X_i, S, Y_j) \ \equiv\ & vars\ (S \cdot Y_j) \subseteq var\ \gamma \cup vars\ X_i \\
\wedge\ & \boxed{\mathrm{this} \notin X_i \cup Y_j} \\
\wedge\ & \mathrm{res} \notin X_i \cup Y_j \\
\wedge\ & distinct\ (X_i\ @\ Y_j)
\end{aligned}$$

Well-formed signatures

$$
\begin{aligned}
\vdash_{mS} (\gamma, X_i, S, Y_j) \quad \equiv \quad & \mathit{vars}\ (S \cdot Y_j) \subseteq \mathit{var}\ \gamma \cup \mathit{vars}\ X_i \\
\wedge \quad & \mathrm{this} \notin X_i \cup Y_j \\
\wedge \quad & \boxed{\mathrm{res} \notin X_i \cup Y_j} \\
\wedge \quad & \mathit{distinct}\ (X_i\ @\ Y_j)
\end{aligned}
$$

Well-formed signatures

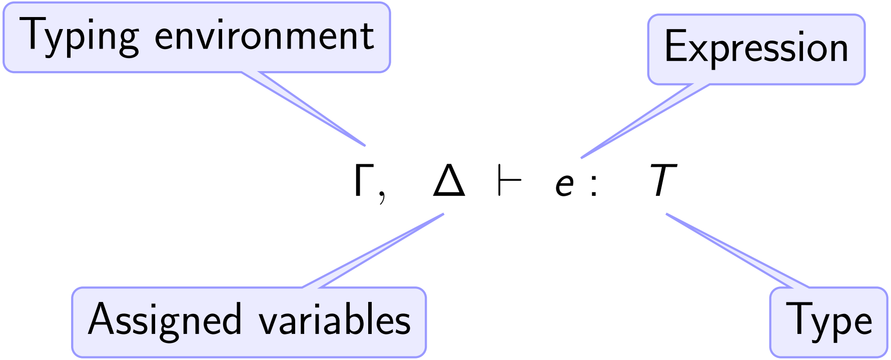
$$\begin{aligned}
\vdash_{mS} (\gamma,\, X_i,\, S,\, Y_j) \;\equiv\;& \mathit{vars}\ (S \cdot Y_j) \subseteq \mathit{var}\ \gamma \cup \mathit{vars}\ X_i \\
\wedge\;& \mathsf{this} \notin X_i \cup Y_j \\
\wedge\;& \mathsf{res} \notin X_i \cup Y_j \\
\wedge\;& \boxed{\mathit{distinct}\ (X_i\ @\ Y_j)}
\end{aligned}$$

Well-formed signatures

$$\begin{array}{rcl} \vdash_{mS} (\gamma, X_i, S, Y_j) & \equiv & \mathit{vars}\ (S \cdot Y_j) \subseteq \mathit{var}\ \gamma \cup \mathit{vars}\ X_i \\ & \wedge & \mathsf{this} \notin X_i \cup Y_j \\ & \wedge & \mathsf{res} \notin X_i \cup Y_j \\ & \wedge & \mathit{distinct}\ (X_i\ @\ Y_j) \end{array}$$

$$\vdash_{cS} (X_i, Y_j) \equiv \ldots$$

Expression typing



$$\Gamma, \quad \Delta \vdash e : \quad T$$

Typing environment

Expression

Assigned variables

Type

Expression typing

$$\frac{x \in \mathit{dom}\ \Gamma \qquad \mathit{nullable}\ (\Gamma\ x) \lor x \in \Delta}{\Gamma, \Delta \vdash x : \Gamma\ x}\ \textsc{TVar}$$

$$\frac{\mathit{nullable}\ T}{\Gamma, \Delta \vdash \mathsf{null} : T}\ \textsc{TNull}$$

x.f			f	
			!	?
	1	committed	1!	1?
x	0	free	◇?	◇?
	◇	unclassified	◇?	◇?

$$\frac{\begin{array}{c}\Gamma, \Delta \vdash x : C^{k_1}! \\ \mathit{fType}\ (C, f) = D n_1 \qquad k_2 = (\mathit{if}\ k_1 = \mathbf{1}\ \mathit{then}\ \mathbf{1}\ \mathit{else}\ \diamond) \\ n_2 = (\mathit{if}\ n_1 = !\ \land\ k_1 = \mathbf{1}\ \mathit{then}\ !\ \mathit{else}\ ?)\end{array}}{\Gamma, \Delta \vdash x\,.\,f : D^{k_2} n_2}\ \textsc{TFld}$$

Statement typing

Typing environment

Statement

$$\Gamma, \quad \Delta \vdash s \mid \Delta', \quad \Sigma$$

Assigned variables

Assigned fields

Statement typing

$$\frac{\Gamma,\,\Delta \vdash e : T \qquad T \sqsubseteq \Gamma\ x}{\Gamma,\,\Delta \vdash x := e \mid \Delta \cup \{x\},\,\varnothing}\ \textsc{TvarAss}$$

$$\frac{\Gamma,\,\Delta \vdash y : C^{k} n_1 \qquad t = D n_2 \qquad D^{k} n_2 \sqsubseteq \Gamma\ x}{\Gamma,\,\Delta \vdash x := (t)\ y \mid \Delta \cup \{x\},\,\varnothing}\ \textsc{TCast}$$

$$\frac{\Gamma,\,\Delta \vdash s_1 \mid \Delta_1,\,\Sigma_1 \qquad \Gamma,\,\Delta_1 \vdash s_2 \mid \Delta_2,\,\Sigma_2}{\Gamma,\,\Delta \vdash s_1;\ s_2 \mid \Delta_2,\,\Sigma_1 \cup \Sigma_2}\ \textsc{TSeq}$$

$$\frac{\begin{array}{c}\Gamma,\,\Delta \vdash x : C^{k_1}! \qquad \mathit{fType}\ (C,\,f) = Dn \\ \Gamma,\,\Delta \vdash y : T \qquad T \sqsubseteq D^{k_2} n \qquad k_1 = \mathbf{0} \vee k_2 = \mathbf{1} \qquad \Sigma = (\mathit{if}\ x = \mathsf{this}\ \mathit{then}\ \{f\}\ \mathit{else}\ \varnothing)\end{array}}{\Gamma,\,\Delta \vdash x{\bullet}f := y \mid \Delta,\,\Sigma}\ \textsc{TfldAss}$$

$$\frac{\begin{array}{c}\Gamma,\,\Delta \vdash y : C^{k}! \qquad \mathit{mSig}\ (C,\,m) = (\gamma,\,X_i,\,S,\,Y_j) \\ \vartheta \in \mathit{instances}\ S\ (C^{\gamma}! \cdot X_i) \qquad \Gamma,\,\Delta \vdash z_i : T_i \qquad T_i \sqsubseteq \vartheta\ X_i \qquad \vartheta\ S \sqsubseteq \Gamma\ x \qquad k \sqsubseteq \vartheta\ \gamma\end{array}}{\Gamma,\,\Delta \vdash x := y{\bullet}m\ (z_i) \mid \Delta \cup \{x\},\,\varnothing}\ \textsc{TCall}$$

$$\frac{\begin{array}{c}\mathit{cSig}\ C = (X_i,\,Y_j) \qquad \vartheta \in \mathit{instances}\ C^{\mathbf{0}}!\ X_i \\ \Gamma,\,\Delta \vdash z_i : T_i \qquad T_i \sqsubseteq \vartheta\ X_i \qquad k = (\mathit{if}\ \mathit{committed}^{*}\ T_i\ \mathit{then}\ \mathbf{1}\ \mathit{else}\ \mathbf{0}) \qquad C^{k}! \sqsubseteq \Gamma\ x\end{array}}{\Gamma,\,\Delta \vdash x := \mathsf{new}\ C\ (z_i) \mid \Delta \cup \{x\},\,\varnothing}\ \textsc{TCreate}$$

Statement typing

$$\frac{\Gamma,\, \Delta \vdash e : T \qquad T \sqsubseteq \Gamma\, x}{\Gamma,\, \Delta \vdash x := e \mid \Delta \cup \{x\},\, \varnothing}\ \textsc{TVarAss}$$

`x = e;`			e	
			?	!
x	?	nullable	✓	✓
	!	non-null	✗	✓

$$\frac{\Gamma,\, \Delta \vdash y : C^{k} n_1 \qquad t = D n_2 \qquad D^{k} n_2 \sqsubseteq \Gamma\, x}{\Gamma,\, \Delta \vdash x := (t)\ y \mid \Delta \cup \{x\},\, \varnothing}\ \textsc{TCast}$$

$$\frac{\Gamma,\, \Delta \vdash s_1 \mid \Delta_1,\, \Sigma_1 \qquad \Gamma,\, \Delta_1 \vdash s_2 \mid \Delta_2,\, \Sigma_2}{\Gamma,\, \Delta \vdash s_1;\ s_2 \mid \Delta_2,\, \Sigma_1 \cup \Sigma_2}\ \textsc{TSeq}$$

$$\frac{\begin{array}{c}\Gamma,\, \Delta \vdash x : C^{k_1}! \qquad fType\ (C,\, f) = Dn \\ \Gamma,\, \Delta \vdash y : T \qquad T \sqsubseteq D^{k_2} n \qquad k_1 = \mathbf{0} \vee k_2 = \mathbf{1} \qquad \Sigma = (if\ x = \mathrm{this}\ then\ \{f\}\ else\ \varnothing)\end{array}}{\Gamma,\, \Delta \vdash x.f := y \mid \Delta,\, \Sigma}\ \textsc{TFldAss}$$

$$\frac{\begin{array}{c}\Gamma,\, \Delta \vdash y : C^{k}! \qquad mSig\ (C,\, m) = (\gamma,\, X_i,\, S,\, Y_j) \\ \vartheta \in instances\ S\ (C^{\gamma}! \cdot X_i) \qquad \Gamma,\, \Delta \vdash z_i : T_i \qquad T_i \sqsubseteq \vartheta\ X_i \qquad \vartheta\ S \sqsubseteq \Gamma\, x \qquad k \sqsubseteq \vartheta\ \gamma\end{array}}{\Gamma,\, \Delta \vdash x := y.m\ (z_i) \mid \Delta \cup \{x\},\, \varnothing}\ \textsc{TCall}$$

$$\frac{\begin{array}{c}cSig\ C = (X_i,\, Y_j) \qquad \vartheta \in instances\ C^{\mathbf{0}}!\ X_i \\ \Gamma,\, \Delta \vdash z_i : T_i \qquad T_i \sqsubseteq \vartheta\ X_i \qquad k = (if\ committed^{*}\ T_i\ then\ \mathbf{1}\ else\ \mathbf{0}) \qquad C^{k}! \sqsubseteq \Gamma\, x\end{array}}{\Gamma,\, \Delta \vdash x := \mathrm{new}\ C\ (z_i) \mid \Delta \cup \{x\},\, \varnothing}\ \textsc{TCreate}$$

Statement typing

$$\frac{\Gamma, \Delta \vdash e : T \qquad T \sqsubseteq \Gamma\ x}{\Gamma, \Delta \vdash x := e \mid \Delta \cup \{x\}, \varnothing}\ \textsc{TVarAss}$$

$$\frac{\Gamma, \Delta \vdash y : C^{k} n_1 \qquad t = D n_2 \qquad D^{k} n_2 \sqsubseteq \Gamma\ x}{\Gamma, \Delta \vdash x := (t)\ y \mid \Delta \cup \{x\}, \varnothing}\ \textsc{TCast}$$

$$\frac{\Gamma, \Delta \vdash s_1 \mid \Delta_1, \Sigma_1 \qquad \Gamma, \Delta_1 \vdash s_2 \mid \Delta_2, \Sigma_2}{\Gamma, \Delta \vdash s_1;\ s_2 \mid \Delta_2, \Sigma_1 \cup \Sigma_2}\ \textsc{TSeq}$$

			y	
`x.f = y;`		**1**	**0**	◇
x	**1**	✓	✗	✗
	0	✓	✓	✓
	◇	✓	✗	✗

$$\frac{\begin{array}{c}\Gamma, \Delta \vdash x : C^{k_1}! \qquad fType\ (C, f) = Dn \\ \Gamma, \Delta \vdash y : T \qquad T \sqsubseteq D^{k_2} n \qquad k_1 = \mathbf{0} \lor k_2 = \mathbf{1} \qquad \Sigma = (if\ x = \mathsf{this}\ then\ \{f\}\ else\ \varnothing)\end{array}}{\Gamma, \Delta \vdash x.f := y \mid \Delta, \Sigma}\ \textsc{TFldAss}$$

$$\frac{\begin{array}{c}\Gamma, \Delta \vdash y : C^{k}! \qquad mSig\ (C, m) = (\gamma, X_i, S, Y_j) \\ \vartheta \in instances\ S\ (C^{\gamma}! \cdot X_i) \qquad \Gamma, \Delta \vdash z_i : T_i \qquad T_i \sqsubseteq \vartheta\ X_i \qquad \vartheta\ S \sqsubseteq \Gamma\ x \qquad k \sqsubseteq \vartheta\ \gamma\end{array}}{\Gamma, \Delta \vdash x := y.m\ (z_i) \mid \Delta \cup \{x\}, \varnothing}\ \textsc{TCall}$$

$$\frac{\begin{array}{c}cSig\ C = (X_i, Y_j) \qquad \vartheta \in instances\ C^{\mathbf{0}}!\ X_i \\ \Gamma, \Delta \vdash z_i : T_i \qquad T_i \sqsubseteq \vartheta\ X_i \qquad k = (if\ committed^{*}\ T_i\ then\ \mathbf{1}\ else\ \mathbf{0}) \qquad C^{k}! \sqsubseteq \Gamma\ x\end{array}}{\Gamma, \Delta \vdash x := \mathsf{new}\ C\ (z_i) \mid \Delta \cup \{x\}, \varnothing}\ \textsc{TCreate}$$

Well-formed methods and constructors

$$\boxed{\vdash_m C,\ m}$$

$$\begin{array}{ll}
\vdash_s \mathit{mBody}\ (C,\ m) & \text{w.f. body} \\
\vdash_{mS} (\gamma,\ X_i,\ S,\ Y_j) & \text{w.f. signature} \\
\neg \mathit{nullable}\ S \longrightarrow \mathit{res} \in \Delta & \text{init. Result} \\
\Gamma_0,\ X_i \cup \{\mathit{this}\} \vdash \mathit{mBody}\ (C,\ m) \mid \Delta,\ \Sigma & \text{w.t. body}
\end{array}$$

Well-formed methods and constructors

$$\boxed{\vdash_m C,\ m}$$

$\vdash_s mBody\ (C,\ m)$	w.f. body
$\vdash_{mS}\ (\gamma,\ X_i,\ S,\ Y_j)$	w.f. signature
$\neg nullable\ S \longrightarrow res \in \Delta$	init. Result
$\Gamma_0,\ X_i \cup \{this\} \vdash mBody\ (C,\ m) \mid \Delta,\ \Sigma$	w.t. body

$$\boxed{\vdash_C C}$$

$\vdash_s cBody\ C$	w.f. body
$\vdash_{cS}\ (X_i,\ Y_j)$	w.f. signature
$\{f \in fields\ C \mid \neg nullable\ (fType\ (C,\ f))\} \subseteq \Sigma$	init. fields
$\Gamma_0,\ X_i \cup \{this\} \vdash cBody\ C \mid \Delta,\ \Sigma$	w.t. body

Well-formed methods and constructors

$\boxed{\vdash_m C,\ m}$

$\vdash_s \mathit{mBody}\ (C,\ m)$ w.f. body

$\vdash_{mS} (\gamma,\ X_i,\ S,\ Y_i)$ w.f. signature

$\boxed{\neg \mathit{nullable}\ S \longrightarrow \mathit{res} \in \Delta}$ init. Result

$\Gamma_0,\ X_i \cup \{\mathit{this}\} \vdash \mathit{mBody}\ (C,\ m) \mid \Delta,\ \Sigma$ w.t. body

$\boxed{\vdash_C C}$

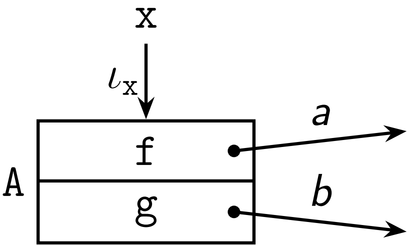
$\vdash_s \mathit{cBody}\ C$ w.f. body

$\vdash_{cS} (X_i,\ Y_i)$ w.f. signature

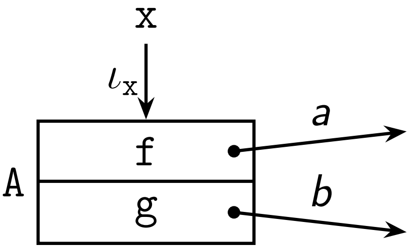
$\boxed{\{f \in \mathit{fields}\ C \mid \neg \mathit{nullable}\ (\mathit{fType}\ (C,\ f))\} \subseteq \Sigma}$ init. fields

$\Gamma_0,\ X_i \cup \{\mathit{this}\} \vdash \mathit{cBody}\ C \mid \Delta,\ \Sigma$ w.t. body

Heap functions

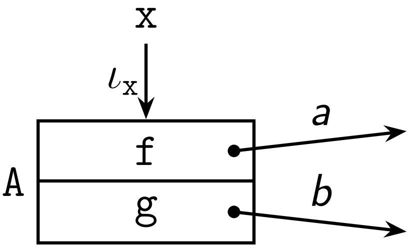


Heap functions



$$h_c : \iota \rightarrow C$$

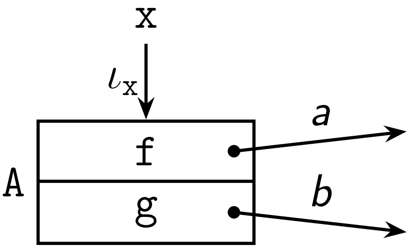
Heap functions



$$h_c : \iota \to C$$

$$h_v : (\iota,\ f) \to \iota$$

Heap functions

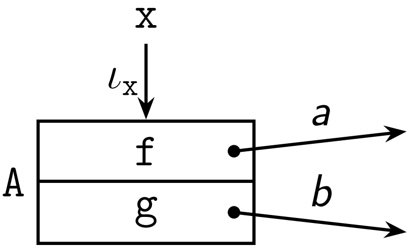


$$h_c : \iota \to C$$

$$h_v : (\iota,\ f) \to \iota$$

$$h \equiv (h_c,\ h_v)$$

Heap functions



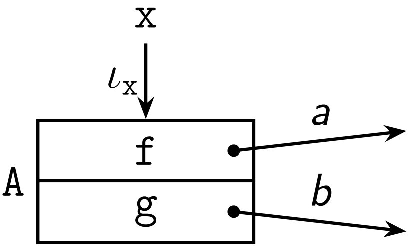
$$h_c : \iota \to C$$

$$h_v : (\iota,\ f) \to \iota$$

$$h \equiv (h_c,\ h_v)$$

$$h_c\ \iota_{\mathtt{x}} = \mathtt{A}$$

Heap functions



$$h_c : \iota \to C$$

$$h_v : (\iota,\ f) \to \iota$$

$$h \equiv (h_c,\ h_v)$$

$$h_c\ \iota_{\mathtt{x}} = \mathtt{A}$$

$$h_v\ (\iota_{\mathtt{x}},\ \mathtt{f}) = a$$
$$h_v\ (\iota_{\mathtt{x}},\ \mathtt{g}) = b$$

Memory allocation

$$\mathit{alloc}\ h\ C = (h',\ \iota)$$

Memory allocation

$$\mathit{alloc}\ h\ C = (h',\ \iota)$$

$$\iota \notin \mathrm{dom}\ h_c$$

Memory allocation

$$alloc\ h\ C = (h',\ \iota)$$

$$\iota \notin \mathrm{dom}\ h_c$$

$$h'_c = h_c\ (\iota \mapsto C)$$

Memory allocation

$$
\begin{aligned}
&\mathit{alloc}\ h\ C = (h',\ \iota)\\
&\qquad \iota \notin \mathrm{dom}\ h_c\\
&\qquad h'_c = h_c\ (\iota \mapsto C)\\
&\qquad h'_v = h_v\ ((\iota, \mathit{fields}\ C) \mapsto \mathit{null})
\end{aligned}
$$

Memory allocation

$$
\begin{aligned}
&\mathit{alloc}\ h\ C = (h',\ \iota)\\
&\qquad \iota \notin \operatorname{dom}\ h_c\\
&\qquad h'_c = h_c\ (\iota \mapsto C)\\
&\qquad h'_v = h_v\ ((\iota, \mathit{fields}\ C) \mapsto \mathit{null})
\end{aligned}
$$

Lemma XYZ. . . . . *Proof.* . . .

Memory allocation

$$
\begin{array}{l}
\mathit{alloc}\ h\ C = (h',\ \iota) \\
\qquad \iota \notin \mathrm{dom}\ h_c \\
\qquad h'_c = h_c\ (\iota \mapsto C) \\
\qquad h'_v = h_v\ ((\iota, \mathit{fields}\ C) \mapsto \mathit{null})
\end{array}
$$

Isabelle/HOL automation

Lemma XYZ. . . . . *Proof.* . . .

Memory allocation

$$
\begin{aligned}
&\mathit{alloc}\ h\ C = (h',\ \iota)\\
&\qquad \iota \notin \mathrm{dom}\ h_c\\
&\qquad h'_c = h_c\ (\iota \mapsto C)\\
&\qquad h'_v = h_v\ ((\iota, \mathit{fields}\ C) \mapsto \mathit{null})
\end{aligned}
$$

Isabelle/HOL automation

Lemma XYZ. . . . . . *Proof.* . . . using *alloc*. □

Memory allocation

$$
\begin{aligned}
&\mathit{alloc}\ h\ C = (h',\ \iota) \\
&\qquad \iota \notin \mathrm{dom}\ h_c \\
&\qquad h'_c = h_c\ (\iota \mapsto C) \\
&\qquad h'_v = h_v\ ((\iota, \mathit{fields}\ C) \mapsto \mathit{null})
\end{aligned}
$$

Isabelle/HOL automation

Lemma XYZ. . . . . . *Proof.* . . . using *alloc*. □

Memory allocation

$$\forall M.\ \forall \iota \in M.$$

$$alloc\ h\ C = (h',\ \iota)$$

$$\iota \notin \mathrm{dom}\ h_c$$

$$h'_c = h_c\ (\iota \mapsto C)$$

$$h'_v = h_v\ ((\iota, fields\ C) \mapsto null)$$

Isabelle/HOL automation

Lemma XYZ. . . . . *Proof.* . . . using *alloc*. □

Memory allocation

$$\forall M.\ \forall \iota \in M.$$

$$\mathit{alloc}\ h\ C = (h',\ \iota)$$

$$\iota \notin \mathrm{dom}\ h_c$$

$$h'_c = h_c\ (\iota \mapsto C)$$

$$h'_v = h_v\ ((\iota, \mathit{fields}\ C) \mapsto \mathit{null})$$

$\mathit{finite}\ M$

A. Summers, P. Müller.
Freedom Before Commitment

Isabelle/HOL automation

Lemma XYZ. . . . . . *Proof.* . . . using *alloc*. □

Memory allocation

$$\forall M.\ \forall \iota \in M.$$

$$\mathit{alloc}\ h\ C = (h',\ \iota)$$

$$\iota \notin \mathrm{dom}\ h_c$$

$$h'_c = h_c\ (\iota \mapsto C)$$

$$h'_v = h_v\ ((\iota, \mathit{fields}\ C) \mapsto \mathit{null})$$

Solutions:

- Out-of-memory exception
- Infinite memory

Lemma XYZ. .... *Proof.* ...

Memory allocation

$$\mathit{alloc}\ h\ C = (h',\ \iota)$$

$$\iota \notin \mathrm{dom}\ h_c$$

$$h'_c = h_c\ (\iota \mapsto C)$$

$$h'_v = h_v\ ((\iota, \mathit{fields}\ C) \mapsto \mathit{null})$$

$$\boxed{\mathit{unbounded}\ h}$$

Lemma XYZ. . . . . *Proof.* . . .

Memory allocation

$$
\begin{aligned}
&\mathit{alloc}\ h\ C = (h',\ \iota)\\
&\qquad \iota \notin \operatorname{dom}\ h_c\\
&\qquad h'_c = h_c\ (\iota \mapsto C)\\
&\qquad h'_v = h_v\ ((\iota, \mathit{fields}\ C) \mapsto \mathit{null})
\end{aligned}
$$

$$\mathit{unbounded}\ h$$

Lemma XYZ. . . . . . *Proof.* . . . using something else. □

Memory allocation

$$\begin{aligned}
&\mathit{alloc}\ h\ C = (h',\ \iota) \\
&\qquad \iota \notin \operatorname{dom}\ h_c \\
&\qquad h'_c = h_c\ (\iota \mapsto C) \\
&\qquad h'_v = h_v\ ((\iota, \mathit{fields}\ C) \mapsto \mathit{null})
\end{aligned}$$

$$\mathit{unbounded}\ h$$

E^{2}F^{2}D^2

Easy
Explanation
Follows
From
Difficult
Discovery

Lemma XYZ. . . . . *Proof.* . . . using something else. □

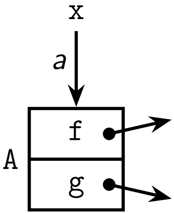
Object structure

$$dom\ h_c = \{\iota \mid \exists\, f.\ (\iota, f) \in dom\ h_v\} \tag{1}$$

A. Summers, P. Müller.
Freedom Before Commitment

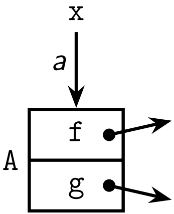
Object structure

$$dom\ h_c = \{\iota \mid \exists f.\ (\iota, f) \in dom\ h_v\} \tag{1}$$



Object structure

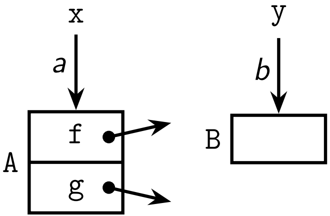
$$dom\ h_c = \{\iota \mid \exists f.\ (\iota, f) \in dom\ h_v\} \tag{1}$$



			fields		
	ι	$h_c\ \iota$	$h_c\ \iota$	$h_v\ \iota$	(1)
x	a	A	f, g	f, g	✓

Object structure

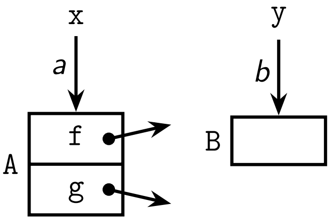
$$dom\ h_c = \{\iota \mid \exists\, f.\ (\iota,\ f) \in dom\ h_v\} \tag{1}$$



	ι	$h_c\ \iota$	*fields*		(1)
			$h_c\ \iota$	$h_v\ \iota$	
x	*a*	A	f, g	f, g	✓

Object structure

$$dom\ h_c = \{\iota \mid \exists\, f.\ (\iota,\ f) \in dom\ h_v\} \tag{1}$$

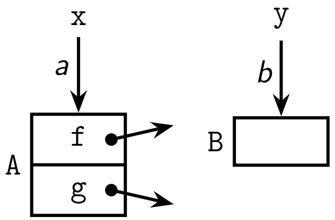


	ι	$h_c\ \iota$	*fields*		(1)
			$h_c\ \iota$	$h_v\ \iota$	
x	a	A	f, g	f, g	✓
y	b	B	$\varnothing$	$\varnothing$	✗

Object structure

$$\mathit{dom}\ h_c = \{\iota \mid \exists\, f.\ (\iota,\ f) \in \mathit{dom}\ h_v\} \tag{1}$$

$$\begin{aligned}\mathit{dom}\ h_c = {} & \{\iota \mid \exists\, f.\ (\iota,\ f) \in \mathit{dom}\ h_v\} \\ & \cup \{\iota \mid \exists\, C.\ h_c\ \iota = C \wedge \mathit{fields}\ C = []\}\end{aligned} \tag{2}$$

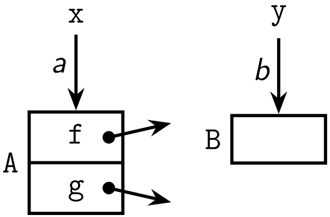


	ι	$h_c\ \iota$	*fields*		(1)
			$h_c\ \iota$	$h_v\ \iota$	
x	a	A	f, g	f, g	✓
y	b	B	$\varnothing$	$\varnothing$	✗

Object structure

$$\mathit{dom}\ h_c = \{\iota \mid \exists f.\ (\iota, f) \in \mathit{dom}\ h_v\} \tag{1}$$

$$\begin{aligned}\mathit{dom}\ h_c = {} & \{\iota \mid \exists f.\ (\iota, f) \in \mathit{dom}\ h_v\} \\ & \cup \{\iota \mid \exists C.\ h_c\ \iota = C \wedge \mathit{fields}\ C = []\}\end{aligned} \tag{2}$$

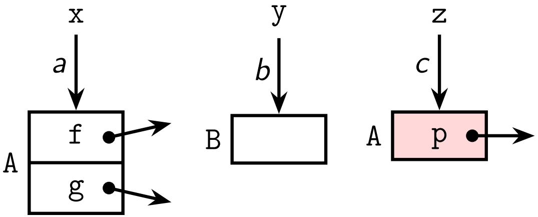


	ι	$h_c\ \iota$	*fields*		(1)	(2)
			$h_c\ \iota$	$h_v\ \iota$		
x	a	A	f, g	f, g	✓	✓
y	b	B	$\varnothing$	$\varnothing$	✗	✓

Object structure

$$\mathit{dom}\ h_c = \{\iota \mid \exists f.\ (\iota, f) \in \mathit{dom}\ h_v\} \tag{1}$$

$$\begin{aligned} \mathit{dom}\ h_c = {} & \{\iota \mid \exists f.\ (\iota, f) \in \mathit{dom}\ h_v\} \\ & \cup \{\iota \mid \exists C.\ h_c\ \iota = C \wedge \mathit{fields}\ C = []\} \end{aligned} \tag{2}$$

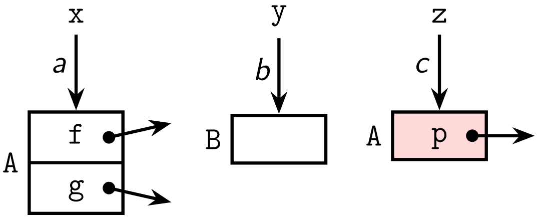


	ι	$h_c\ \iota$	*fields*		(1)	(2)
			$h_c\ \iota$	$h_v\ \iota$		
x	a	A	f, g	f, g	✓	✓
y	b	B	∅	∅	✗	✓

Object structure

$$\mathit{dom}\ h_c = \{\iota \mid \exists\, f.\ (\iota, f) \in \mathit{dom}\ h_v\} \tag{1}$$

$$\begin{aligned}\mathit{dom}\ h_c = {} & \{\iota \mid \exists\, f.\ (\iota, f) \in \mathit{dom}\ h_v\} \\ & \cup \{\iota \mid \exists\, C.\ h_c\ \iota = C \wedge \mathit{fields}\ C = []\}\end{aligned} \tag{2}$$



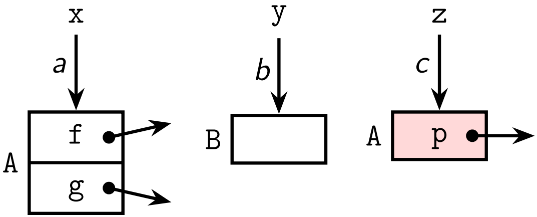
			fields			
	ι	$h_c\ \iota$	$h_c\ \iota$	$h_v\ \iota$	(1)	(2)
x	a	A	f, g	f, g	✓	✓
y	b	B	∅	∅	✗	✓
z	c	A	f, g	p	✗	✗

Object structure

$$\mathit{dom}\ h_c = \{\iota \mid \exists\, f.\ (\iota, f) \in \mathit{dom}\ h_v\} \tag{1}$$

$$\begin{aligned}\mathit{dom}\ h_c = {} & \{\iota \mid \exists\, f.\ (\iota, f) \in \mathit{dom}\ h_v\} \\ & \cup \{\iota \mid \exists\, C.\ h_c\ \iota = C \wedge \mathit{fields}\ C = []\}\end{aligned} \tag{2}$$

$$\mathit{dom}\ h_v = \{(\iota, f) \mid \iota \in \mathit{dom}\ h_c \wedge f \in \mathit{fields}\ (h_c\ \iota)\} \tag{3}$$



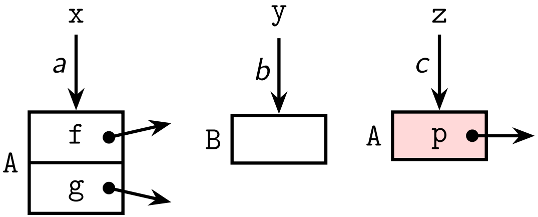
	ι	$h_c\ \iota$	*fields*		(1)	(2)
			$h_c\ \iota$	$h_v\ \iota$		
x	a	A	f, g	f, g	✓	✓
y	b	B	$\varnothing$	$\varnothing$	✗	✓
z	c	A	f, g	p	✗	✗

Object structure

$$\mathit{dom}\ h_c = \{\iota \mid \exists\, f.\ (\iota,\, f) \in \mathit{dom}\ h_v\} \tag{1}$$

$$\begin{aligned}\mathit{dom}\ h_c = {} & \{\iota \mid \exists\, f.\ (\iota,\, f) \in \mathit{dom}\ h_v\} \\ & \cup \{\iota \mid \exists\, C.\ h_c\ \iota = C \wedge \mathit{fields}\ C = []\}\end{aligned} \tag{2}$$

$$\mathit{dom}\ h_v = \{(\iota,\, f) \mid \iota \in \mathit{dom}\ h_c \wedge f \in \mathit{fields}\ (h_c\ \iota)\} \tag{3}$$



			fields				
	ι	$h_c\ \iota$	$h_c\ \iota$	$h_v\ \iota$	(1)	(2)	(3)
x	a	A	f, g	f, g	✓	✓	✓
y	b	B	∅	∅	✗	✓	✓
z	c	A	f, g	p	✗	✗	✓

Object structure

$$\mathit{dom}\ h_c = \{\iota \mid \exists\, f.\ (\iota,\ f) \in \mathit{dom}\ h_v\} \tag{1}$$

$$\begin{aligned}\mathit{dom}\ h_c = {} & \{\iota \mid \exists\, f.\ (\iota,\ f) \in \mathit{dom}\ h_v\} \\ & \cup \{\iota \mid \exists\, C.\ h_c\ \iota = C \wedge \mathit{fields}\ C = [\,]\}\end{aligned} \tag{2}$$

$$\mathit{dom}\ h_v = \{(\iota,\ f) \mid \iota \in \mathit{dom}\ h_c \wedge f \in \mathit{fields}\ (h_c\ \iota)\} \tag{3}$$

$\vdash^0{}_{\mathit{dom}}\ h \equiv (2)$ weak

Object structure

$$\mathit{dom}\ h_c = \{\iota \mid \exists\, f.\ (\iota,\ f) \in \mathit{dom}\ h_v\} \tag{1}$$

$$\begin{aligned}\mathit{dom}\ h_c = {} & \{\iota \mid \exists\, f.\ (\iota,\ f) \in \mathit{dom}\ h_v\} \\ & \cup \{\iota \mid \exists\, C.\ h_c\ \iota = C \wedge \mathit{fields}\ C = []\}\end{aligned} \tag{2}$$

$$\mathit{dom}\ h_v = \{(\iota,\ f) \mid \iota \in \mathit{dom}\ h_c \wedge f \in \mathit{fields}\ (h_c\ \iota)\} \tag{3}$$

$\vdash^0{}_{dom}\ h \equiv (2)$ weak

$\vdash_{dom}\ h \equiv (3)$ strong

Object structure

$$\mathit{dom}\ h_c = \{\iota \mid \exists\, f.\ (\iota, f) \in \mathit{dom}\ h_v\} \tag{1}$$

$$\begin{aligned} \mathit{dom}\ h_c = {} & \{\iota \mid \exists\, f.\ (\iota, f) \in \mathit{dom}\ h_v\} \\ & \cup \{\iota \mid \exists\, C.\ h_c\ \iota = C \wedge \mathit{fields}\ C = \mathit{[]}\} \end{aligned} \tag{2}$$

$$\mathit{dom}\ h_v = \{(\iota, f) \mid \iota \in \mathit{dom}\ h_c \wedge f \in \mathit{fields}\ (h_c\ \iota)\} \tag{3}$$

$\vdash^0{}_{\mathit{dom}}\ h \equiv (2)$ weak

$\vdash_{\mathit{dom}}\ h \equiv (3)$ strong

$$\vdash_{\mathit{dom}}\ h \Longrightarrow \vdash^0{}_{\mathit{dom}}\ h$$

Object structure

$$\mathit{dom}\ h_c = \{\iota \mid \exists\, f.\ (\iota, f) \in \mathit{dom}\ h_v\} \tag{1}$$

$$\begin{aligned}\mathit{dom}\ h_c = {} & \{\iota \mid \exists\, f.\ (\iota, f) \in \mathit{dom}\ h_v\} \\ & \cup \{\iota \mid \exists\, C.\ h_c\ \iota = C \wedge \mathit{fields}\ C = \mathit{[]}\}\end{aligned} \tag{2}$$

$$\mathit{dom}\ h_v = \{(\iota, f) \mid \iota \in \mathit{dom}\ h_c \wedge f \in \mathit{fields}\ (h_c\ \iota)\} \tag{3}$$

$\vdash^0{}_{\mathit{dom}}\ h \equiv (2)$ weak

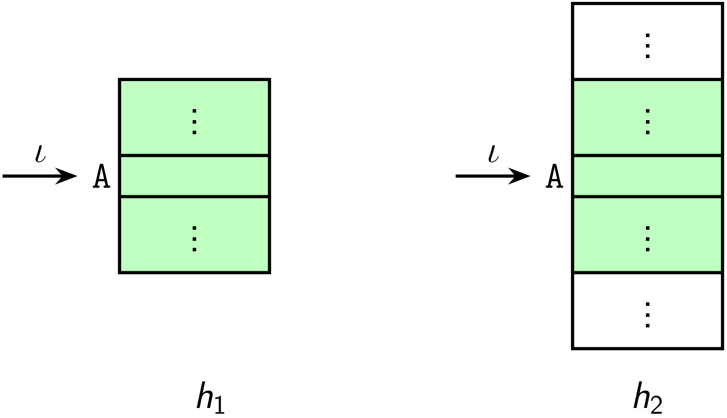
$\vdash_{\mathit{dom}}\ h \equiv (3)$ strong

$$\vdash_{\mathit{dom}}\ h \Longrightarrow \vdash^0{}_{\mathit{dom}}\ h$$

$$\vdash_{\mathit{im}}\ h \equiv h_v\ `\ \mathit{dom}\ h_v \subseteq \{\mathrm{null}\} \cup \{\iota \mid \iota \in \mathit{dom}\ h_c\}$$

Heap ordering

$$h_1 \leq h_2 \equiv dom\ h_{1c} \subseteq dom\ h_{2c} \wedge (\forall\, \iota \in dom\ h_{1c}.\ h_{2c}\ \iota = h_{1c}\ \iota)$$



Expression value

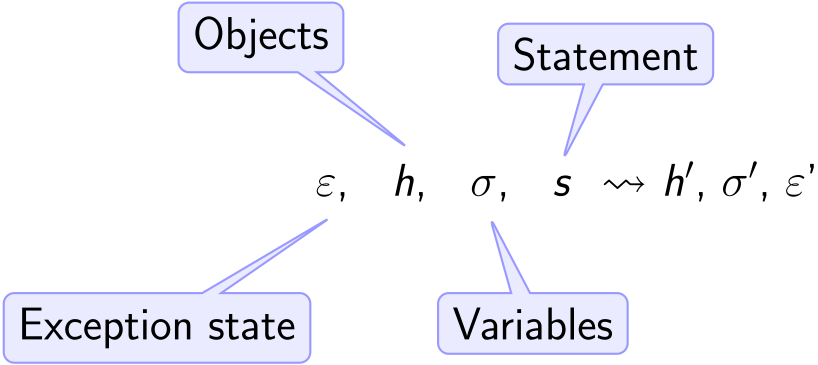
$$
\begin{aligned}
\lfloor x \rfloor_{h,\sigma} &= \sigma \ x \\
\lfloor \mathsf{null} \rfloor_{h,\sigma} &= \mathsf{null} \\
\lfloor x \, . \, f \rfloor_{h,\sigma} &= \begin{cases} h_v \ (\sigma \ x, \ f) & \text{if } \sigma \ x \neq \mathsf{null} \\ \mathsf{derefExc} & \text{if } \sigma \ x = \mathsf{null} \end{cases}
\end{aligned}
$$

Operational semantics

Exception state ε:

- ok
- castExc
- derefExc

Operational semantics



$$\varepsilon, \quad h, \quad \sigma, \quad s \leadsto h', \sigma', \varepsilon'$$

Objects

Statement

Exception state

Variables

Operational semantics

$$\frac{\lfloor e \rfloor_{h,\sigma} = v}{\mathsf{ok},\ h,\ \sigma,\ x := e \rightsquigarrow h,\ \sigma(x \mapsto v),\ \mathsf{ok}}\ \textsc{varAss} \qquad \frac{\lfloor e \rfloor_{h,\sigma} = \mathsf{derefExc}}{\mathsf{ok},\ h,\ \sigma,\ x := e \rightsquigarrow h,\ \sigma,\ \mathsf{derefExc}}\ \textsc{varAssBad}$$

$$\frac{\sigma\ x = \iota}{\mathsf{ok},\ h,\ \sigma,\ x.f\ := y \rightsquigarrow (h_v((\iota,\ f) \mapsto \sigma\ y),\ h_c),\ \sigma,\ \mathsf{ok}}\ \textsc{fldAss} \qquad \frac{\sigma\ x = \mathsf{null}}{\mathsf{ok},\ h,\ \sigma,\ x.f\ := y \rightsquigarrow h,\ \sigma,\ \mathsf{derefExc}}\ \textsc{fldAssBad}$$

$$\frac{\sigma\ y = \mathsf{null}}{\mathsf{ok},\ h,\ \sigma,\ x := y.m\ (z_i) \rightsquigarrow h,\ \sigma,\ \mathsf{derefExc}}\ \textsc{callBad}$$

$$\frac{\begin{array}{ccc} \sigma\ y = \iota & h_c\ \iota = C & mSig\ (C,\ m) = (\gamma,\ X_i,\ S,\ Y_j) \\ \sigma_1\ = [\mathsf{this} \mapsto \iota,\ X_i \mapsto \sigma\ z_i,\ \mathsf{res} \mapsto \mathsf{null},\ Y_j \mapsto \mathsf{null}] & mBody\ (C,\ m) = s & \mathsf{ok},\ h,\ \sigma_1,\ s \rightsquigarrow h',\ \sigma',\ \varepsilon \end{array}}{\mathsf{ok},\ h,\ \sigma,\ x := y.m\ (z_i) \rightsquigarrow h',\ \sigma(x \mapsto \sigma'\ \mathsf{res}),\ \varepsilon}\ \textsc{call}$$

$$\frac{\begin{array}{ccc} cSig\ C = (X_i,\ Y_j) & alloc\ h\ C = (h_1,\ \iota_1) & \\ \sigma_1\ = [\mathsf{this} \mapsto \iota_1,\ X_i \mapsto \sigma\ z_i,\ Y_j \mapsto \mathsf{null}] & cBody\ C = s & \mathsf{ok},\ h_1,\ \sigma_1,\ s \rightsquigarrow h',\ \sigma_2,\ \varepsilon \end{array}}{\mathsf{ok},\ h,\ \sigma,\ x := \mathsf{new}\ C\ (z_i) \rightsquigarrow h',\ \sigma(x \mapsto \iota_1),\ \varepsilon}\ \textsc{create}$$

$$\frac{h \vdash \sigma\ y : t}{\mathsf{ok},\ h,\ \sigma,\ x := (t)\ y \rightsquigarrow h,\ \sigma(x \mapsto \sigma\ y),\ \mathsf{ok}}\ \textsc{cast} \qquad \frac{\neg\ h \vdash \sigma\ y : t}{\mathsf{ok},\ h,\ \sigma,\ x := (t)\ y \rightsquigarrow h,\ \sigma,\ \mathsf{castExc}}\ \textsc{castBad}$$

$$\frac{\mathsf{ok},\ h,\ \sigma,\ s_1 \rightsquigarrow h_1,\ \sigma_1,\ \mathsf{ok} \qquad \mathsf{ok},\ h_1,\ \sigma_1,\ s_2 \rightsquigarrow h_2,\ \sigma_2,\ \varepsilon}{\mathsf{ok},\ h,\ \sigma,\ s_1;\ s_2 \rightsquigarrow h_2,\ \sigma_2,\ \varepsilon}\ \textsc{seq} \qquad \frac{\mathsf{ok},\ h,\ \sigma,\ s_1 \rightsquigarrow h_1,\ \sigma_1,\ \varepsilon \qquad \varepsilon \neq \mathsf{ok}}{\mathsf{ok},\ h,\ \sigma,\ s_1;\ s_2 \rightsquigarrow h_1,\ \sigma_1,\ \varepsilon}\ \textsc{seqBad}$$

Reachability

$$\mathit{reaches}_1\ h_v \equiv \lambda \iota_1\ \iota_2.\ \exists\, f.\ h_v\ (\iota_1,\ f) = \iota_2$$

Reachability

$$\mathit{reaches}_1\ h_v \equiv \lambda \iota_1\ \iota_2.\ \exists\, f.\ h_v\ (\iota_1,\ f) = \iota_2$$

$$\mathit{reaches}\ h_v \equiv (\mathit{reaches}_1\ h_v)^{**}$$

Reachability

$$\textit{reaches}_1\ h_v \equiv \lambda\iota_1\ \iota_2.\ \exists f.\ h_v\ (\iota_1, f) = \iota_2$$

$$\textit{reaches}\ h_v \equiv (\textit{reaches}_1\ h_v)^{**}$$

$$\textit{reaches}\ h_v\ V\ v \equiv \exists\, v' {\in} V.\ \textit{reaches}\ h_v\ v'\ v$$
$$\textit{reaches}\ h_v\ v\ V \equiv \exists\, v' {\in} V.\ \textit{reaches}\ h_v\ v\ v'$$

Initialization

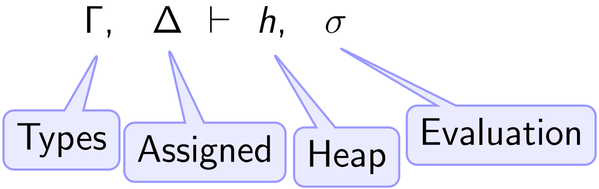
$$\begin{aligned} \mathit{init}\ h\ \iota \equiv \forall f \in \mathit{fields}\ (h_c\ \iota).&\\ \neg\ \mathit{nullable}\ (\mathit{fType}\ (h_c\ \iota,\ f)) &\longrightarrow h_v\ (\iota,\ f) \neq \mathsf{null} \end{aligned}$$

Initialization

$$\begin{aligned}\mathit{init}\ h\ \iota \equiv \forall f \in \mathit{fields}\ (h_c\ \iota).&\\ \neg\ \mathit{nullable}\ (\mathit{fType}\ (h_c\ \iota,\ f)) \longrightarrow h_v\ (\iota,\ f) \neq \mathrm{null}&\end{aligned}$$

$$\mathit{deep\_init}\ h\ \iota \equiv \forall\ \iota'.\ \mathit{reaches}\ h_v\ \iota\ \iota' \longrightarrow \mathit{init}\ h\ \iota'$$

Run-time configuration



$$\Gamma, \quad \Delta \vdash h, \quad \sigma$$

Types

Assigned

Heap

Evaluation

Run-time configuration

$$
\begin{aligned}
&\Gamma,\ \ \Delta \vdash h,\ \ \sigma \ \equiv \\
&\quad \mathit{dom}\ \sigma = \mathit{dom}\ \Gamma \wedge \mathrm{this} \in \mathit{dom}\ \sigma \\
&\wedge \forall \iota{\in}\mathit{dom}\ h_c.\ \forall f{\in}\mathit{fields}\ (h_c\ \iota).\ h_v\ (\iota,\ f) \neq \mathrm{null} \longrightarrow \\
&\quad h_v\ (\iota,\ f) \in \mathit{dom}\ h_c \wedge h_c\ (h_v\ (\iota,\ f)) \sqsubseteq \mathit{fType}\ (h_c\ \iota,\ f) \\
&\wedge \forall x{\in}\mathit{dom}\ \sigma.\ \neg\ \mathit{nullable}\ (\Gamma\ x) \wedge x \in \Delta \longrightarrow \sigma\ x \neq \mathrm{null} \\
&\wedge \forall x{\in}\mathit{dom}\ \sigma.\ \sigma\ x \neq \mathrm{null} \longrightarrow \sigma\ x \in \mathit{dom}\ h_c \wedge h \vdash \sigma\ x : \Gamma\ x \\
&\wedge \forall x{\in}\mathit{dom}\ \sigma.\ \forall y{\in}\mathit{dom}\ \sigma.\ \mathit{committed}\ (\Gamma\ x) \longrightarrow \\
&\quad \mathit{deep\_init}_v\ h\ (\sigma\ x) \wedge (\mathit{free}\ (\Gamma\ y) \longrightarrow \neg\ \mathit{reaches}\ h_v \\
&\quad (\sigma\ x)\ (\sigma\ y)) \\
&\wedge \neg\ \mathit{nullable}\ (\Gamma\ \mathrm{this}) \wedge \mathrm{this} \in \Delta \\
&\wedge \vdash_{\mathit{dom}}\ h \\
&\wedge \vdash_{\mathit{im}}\ h
\end{aligned}
$$

Run-time configuration

$$
\begin{aligned}
&\Gamma,\ \ \Delta \vdash h,\ \ \sigma \ \ \equiv \\
&\qquad \mathit{dom}\ \sigma = \mathit{dom}\ \Gamma \wedge \mathrm{this} \in \mathit{dom}\ \sigma \\
&\quad \wedge \forall\, \iota \in \mathit{dom}\ h_c.\ \forall\, f \in \mathit{fields}\ (h_c\ \iota).\ h_v\ (\iota,\ f) \neq \mathrm{null} \longrightarrow \\
&\qquad h_v\ (\iota,\ f) \in \mathit{dom}\ h_c \wedge h_c\ (h_v\ (\iota,\ f)) \sqsubseteq \mathit{fType}\ (h_c\ \iota,\ f) \\
&\quad \wedge \forall\, x \in \mathit{dom}\ \sigma.\ \neg\ \mathit{nullable}\ (\Gamma\ x) \wedge x \in \Delta \longrightarrow \sigma\ x \neq \mathrm{null} \\
&\quad \wedge \forall\, x \in \mathit{dom}\ \sigma.\ \sigma\ x \neq \mathrm{null} \longrightarrow \sigma\ x \in \mathit{dom}\ h_c \wedge h \vdash \sigma\ x : \Gamma\ x \\
&\quad \wedge \forall\, x \in \mathit{dom}\ \sigma.\ \forall\, y \in \mathit{dom}\ \sigma.\ \mathit{committed}\ (\Gamma\ x) \longrightarrow \\
&\qquad \mathit{deep\_init}_v\ h\ (\sigma\ x) \wedge (\mathit{free}\ (\Gamma\ y) \longrightarrow \neg\ \mathit{reaches}\ h_v \\
&\qquad (\sigma\ x)\ (\sigma\ y)) \\
&\quad \boxed{\begin{aligned}
&\wedge \neg\ \mathit{nullable}\ (\Gamma\ \mathrm{this}) \wedge \mathrm{this} \in \Delta \\
&\wedge \vdash_{\mathit{dom}}\ h \\
&\wedge \vdash_{\mathit{im}}\ h
\end{aligned}}
\end{aligned}
$$

Run-time configuration

$$
\begin{aligned}
&\Gamma,\quad \Delta \vdash h,\quad \sigma \quad \equiv\\
&\qquad \mathit{dom}\ \sigma = \mathit{dom}\ \Gamma \land \mathsf{this} \in \mathit{dom}\ \sigma\\
&\quad \land \forall\, \iota{\in}\mathit{dom}\ h_c.\ \forall\, f{\in}\mathit{fields}\ (h_c\ \iota).\ h_v\ (\iota,\ f) \neq \mathsf{null} \longrightarrow\\
&\qquad h_v\ (\iota,\ f) \in \mathit{dom}\ h_c \land h_c\ (h_v\ (\iota,\ f)) \sqsubseteq \mathit{fType}\ (h_c\ \iota,\ f)\\
&\quad \land \forall\, x{\in}\mathit{dom}\ \sigma.\ \neg\ \mathit{nullable}\ (\Gamma\ x) \land x \in \Delta \longrightarrow \sigma\ x \neq \mathsf{null}\\
&\quad \land \forall\, x{\in}\mathit{dom}\ \sigma.\ \sigma\ x \neq \mathsf{null} \longrightarrow \sigma\ x \in \mathit{dom}\ h_c \land h \vdash \sigma\ x : \Gamma\ x\\
&\quad \land \forall\, x{\in}\mathit{dom}\ \sigma.\ \forall\, y{\in}\mathit{dom}\ \sigma.\ \mathit{committed}\ (\Gamma\ x) \longrightarrow\\
&\qquad \mathit{deep\_init}_v\ h\ (\sigma\ x) \land (\mathit{free}\ (\Gamma\ y) \longrightarrow \neg\ \mathit{reaches}\ h_v\\
&\qquad (\sigma\ x)\ (\sigma\ y))\\
&\quad \boxed{\begin{aligned}
&\land \neg\ \mathit{nullable}\ (\Gamma\ \mathsf{this}) \land \mathsf{this} \in \Delta\\
&\land \vdash_{\mathit{dom}}\ h\\
&\land \vdash_{\mathit{im}}\ h
\end{aligned}}
\end{aligned}
$$

$E^2F^2D^2$

Safety and preservation

Theorem. If all the following is true

$\Gamma, \Delta \vdash h, \sigma$

$\Gamma, \Delta \vdash s \mid \Delta', \Sigma$

$\text{ok}, h, \sigma, s \rightsquigarrow h', \sigma', \varepsilon$

$\varepsilon \neq \text{castExc}$

$\vdash_s s$

unbounded h

then program s finishes in good configuration $\Gamma, \Delta' \vdash h', \sigma'$ and does not dereference a null pointer: $\varepsilon = \text{ok}$.

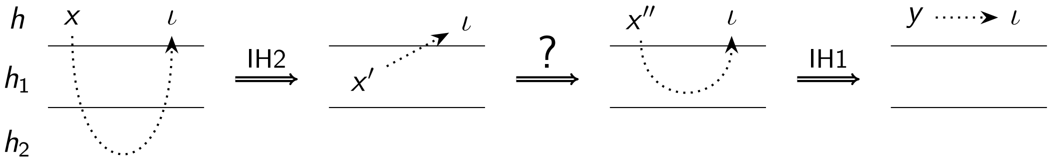
Induction goals

Version	Ind. goals	Instructions		Min. cases
		“good”	“bad”	
Original	14	6	5	89
Verified	20			125

Reachability goal

$$\forall\, \iota{\in}\mathit{dom}\ h'_c.\ \forall\, x{\in}\mathit{dom}\ \sigma'.\ \mathit{reaches}\ h'_v\ (\sigma'\ x)\ \iota \wedge \mathit{committed}\ (\Gamma\ x) \wedge$$
$$\iota \in \mathit{dom}\ h_c \wedge \sigma'\ x \in \mathit{dom}\ h_c \longrightarrow (\exists\, y{\in}\mathit{dom}\ \sigma.\ \mathit{committed}\ (\Gamma\ y)$$
$$\wedge\ \mathit{reaches}\ h_v\ (\sigma\ y)\ \iota)$$

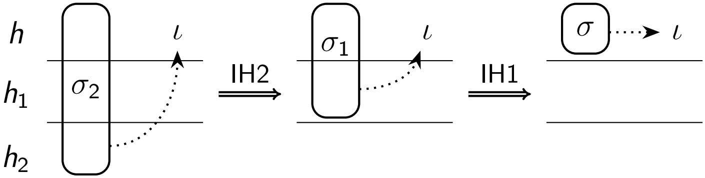
$s_1;\ s_2$



Reachability goal

$$\forall\, \iota {\in} \mathit{dom}\ h_c.\ \mathit{reaches}\ h'_v\ (\sigma'\ `\ \{x \in \mathit{dom}\ \sigma' \mid \mathit{committed}\ (\Gamma\ x)\})\ \iota \longrightarrow \\ \mathit{reaches}\ h_v\ (\sigma\ `\ \{x \in \mathit{dom}\ \sigma \mid \mathit{committed}\ (\Gamma\ x)\})\ \iota$$

$s_1;\ s_2$



Statistics

Publication	Proof size	Correct?
OOPSLA'11	0	N/A
ETH TR'10	$\approx$ 20 pages	No
this	$\approx$ 200 pages	Isabelle/HOL

Result

expr.method (args);

~~*tmp* = **expr**;~~
~~*if (tmp == null)*~~
        ~~*throw new NullPointerException ();*~~
~~*else*~~
        ~~*tmp*.**method(args);**~~